OIPE
NOV 30 2005
PATENT & TRADEMARK OFFICE
IAP88

Docket No.: 042390.P8629X

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Carl M. Ellison Application No.: 09/672,602 Filed: September 29, 2000 For: ATTESTATION KEY MEMORY DEVICE AND BUS	Examiner: Kaveh Abrishamkar Art Group: 2131

RESPONSE TO NOTIFICATION OF
NON-COMPLIANT APPEAL BRIEF

(37 CFR § 41.37)

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

Applicants are filing a new Appeal Brief in response to the Notification of Non-Compliant Appeal Brief dated August 24, 2005.

REMARK

Applicants are filing a new Appeal Brief in response to the Notification of Non-Compliant Appeal Brief dated August 24, 2005.

The Notification states that the Appeal Brief filed on June 13, 2005 is defective for failure to comply with one or more provisions of 37 CFR § 41.37. In particular, the Notification states that the brief does not contain a concise explanation of the subject matter defined in each of the independent claims involved in the appeal. The Notification further states that "[t]he copying and pasting of the specification into the body of the "Summary of Invention" section, is not believed to conform with the MPEP outlines of a 'concise explanation of the invention defined in the claims involved in the appeal." In response, Applicants have revised the "Summary of Claimed Subject Matter" accordingly.

In addition, Applicants have included section X and XI regarding Evidence Appendix and Related Proceedings Appendix, respectively.

Applicants believe that the new Appeal Brief complies with the requirements of 37 CFR § 41.37 and request the new Appeal Brief be forwarded to the Board of Patent Appeals and Interferences for consideration.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: November 25, 2005

THINH V. NGUYEN
Reg. No. 42,034

12400 Wilshire Blvd., 7th Floor
Los Angeles, CA 90025-1026
(714) 557-3800

OIPE IAP88
NOV 30 2005
PATENT & TRADEMARK OFFICE

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Carl M. Ellison Application No.: 09/672,602 Filed: September 29, 2000 For: ATTESTATION KEY MEMORY DEVICE AND BUS	Examiner: Kaveh Abrishamkar Art Group: 2131

APPEAL BRIEF

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

Applicants submit the following Appeal Brief pursuant to 37 C.F.R. § 41.37 for consideration by the Board of Patent Appeals and Interferences. Applicants also submit herewith our check number 0164 in the amount of $450.00 to cover the cost of extension of time for two months as required by 37 C.F.R. § 1.136. Please charge any additional fees or credit any overpayment to our deposit Account No. 02-2666. A duplicate copy of the Fee Transmittal is enclosed for this purpose.

OIPE IAP88
NOV 30 2005
PATENT & TRADEMARK OFFICE

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

The real party in interest is the assignee, Intel Corporation.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences known to the appellants, the appellants' legal representative, or assignee, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 1-80 are pending in the application. Claims 1-80 of the present application remain rejected. The Applicants hereby appeal the rejection of claims 1-80.

IV. STATUS OF AMENDMENTS

The Applicants filed an amendment on February 28, 2005, in response to a Final Office Action issued by the Examiner on January 11, 2005. In response to the February 28, 2005 amendment, the Examiner issued an Advisory Action on March 17, 2005. The Applicants filed a Notice of Appeal on April 11, 2005.

V. SUMMARY OF CLAIMED SUBJECT MATTER

1. Independent claims 1, 21, 41, and 61:

One embodiment of the present invention is a technique to perform remote attestation by a an attestation key memory (AKM) device to attest an isolated execution mode[1] and to prove validity of a program loaded into an isolated memory area using an isolated digest stored in a digest memory[2].

A processor 110 is interfaced to a memory controller 130, a system memory 140, an input/output controller chipset 150, and an AKM device 186[3]. The processor 110 operates in a normal execution mode and an isolated execution mode[4]. The chipset 150

[1] See Specification, page 16, lines 9-15.
[2] See Specification, page 14, lines 6-13.
[3] See Specification page 9, lines 10-16; Figure 1C.
[4] See Specification, page 10, lines 3-10.

stores a digest memory 154. The digest memory has an isolated digest in a secure environment[5]. The secure environment is associated with an isolated memory area 70 accessible by the processor 110[6]. The isolate digest may include a processor nub loader digest, a processor nub digest, and optionally additional digests. It is used to attest the state of the current isolated execution and to prove the validity of the software loaded into the isolated area[7]. The remote attestation is performed by an AKM device 186 with respect to a memory controller 130 and the chipset 150[8].

2. Dependent claims 6-19, 26-39, 46-59, and 66-79:

a) Dependent claims 6, 26, 46, and 66:

A token bus interface 159 includes an interface 210, a communication storage 220, and a chipset storage 270. The communication storage 220 includes a configuration storage 230, a status register 240, a command register 250, and an input/output block (IOB) 260. The configuration storage 230 stores configuration information 232. The status register 240 stores device status 242. The command register 250 stores device command 252. The IOB 260 stored input data 262 and output data 264[9].

b) Dependent claims 7-9, 27-29, 47-49, and 67-69:

The configuration storage 230 includes a manufacturer identifier 310, a revision identifier 320, an interface set identifier 330, a static public key 340, and a static key certificate 350[10]. The manufacturer identifier 310 identifies the manufacturer of the AKM device 186. The revision identifier 320 provides a revision number of the AKM device 186. The interface set identifier 330 identifies the interface set that is supported by the device 186. The static public key 340 is a public key with a short key identification. The key certificate 350 is a key certificate with a short key identification[11]. The interface set identified by the interface set identifier 330 identifies may include an initialization set 360, an attestation set 370, and a device interface set 380. For a typical remote attestation, the initialization set 360 is needed. The initialization set 360 may be hard-coded and is used to reset and initialize the device. The initialization set 360 includes an idle state 362, a reset command 364, a connect command 366, and a reserved operation 368. The reset

[5] See Specification, page 13, lines 4-7; Figure 1C.
[6] See Specification, page 12, lines 6-8, lines 12-16; Figures 1B and 1C.
[7] See Specification, page 14, lines 6-13.
[8] See Specification, page 16, lines 10-13.
[9] See Specification page 18, lines 5-10; Figure 2 (reference 220).
[10] See Specification page 18, lines 20-24; Figure 3.

command 364 causes the device to reset and perform a self-test operation. The connect command 366 sets the connect bit in the status register 240[12].

c) Dependent claims 10, 13-14, 30, 33-34, 50, 53-54, 70, and 73-74:

The attestation set 370 includes a signing operation 372, a public key enumeration 374, and a key certificate enumeration 376. The signing operation 372 provides the remote attestation to verify the validity of the platform running a particular software in the secure environment. The public key enumeration 374 enumerates any additional public keys that are not part of the static configuration information 232. The key certificate enumeration 376 enumerates any additional key certificates that are not part of the static configuration information 232[13].

d) Dependent claims 11-12, 31-32, 51-52, and 71-72:

The status register 240 includes a self-test field 510, a connection field 520, an estimate field 530, and a reserved field 540. The self-test field 510 provides a result of the self-test operation in response to the reset command. The connection field 520 indicates that the device is responsive to the connect command. The estimate field 530 provides an estimate in some time unit (e.g., milliseconds) to indicate how long a current operation is expected to take[14].

e) Dependent claims 15-16, 35-36, 55-56, and 75-76:

The signing operation 372 includes a hash function 410 and a cryptographic function 420. The hash function 410 performs hashing on a parameter in the chipset storage 270 such as the processor nub loader hash 272, the chipset hash log 274, the software hash 276, and the nonce 278. The result of this hashing operation is then encrypted by the cryptographic function 420 using the private key 280 stored in the chipset[15].

f) Dependent claims 17-19, 37-39, 57-59, and 77-79:

The chipset storage 270 includes a processor nub loader hash 272, a chipset hash log 274, a software hash 276, and a nonce 278. The processor nub loader hash 272 and

---

[11] See Specification page 19, lines 1-6.
[12] See Specification page 19, lines 7-19.
[13] See Specification page 10, lines 20-27.
[14] See Specification page 20, lines 24-27; page 21, lines 1-9; Figure 5.
[15] See Specification page 20, lines 14-23; Figure 4.

the chipset hash log 274 can be read directly by the AKM device 186. The software hash 276 and the nonce 278 are provided by the processor nub 18[16].

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Claims 1-5, 20-25, 40-45, 60-65 and 80 stand rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 6,327,652 issued to England et al. ("England").
2. Claims 6-19, 26-39, 46-59, and 66-79 stand rejected under 35 U.S.C. §103(a) as being unpatentable over England in view of U.S. Patent No. 4,319,323 issued to Ermolovich ("Ermolovich").

VII. ARGUMENTS

A. Claims 1-5, 20-25, 40-45, 60-65 and 80 Are Not Anticipated by England.

In the Final Office Action, the Examiner rejected claims 1-5, 20-25, 40-45, 60-65 and 80 under 35 U.S.C. §102(e) as being anticipated U.S. Patent No. 6,327,652 issued to England et al. ("England"). Applicants respectfully traverse the rejection and contend that the Examiner has not met the burden of establishing a prima facie case of anticipation.

To anticipate a claim, the reference must teach every element of the claim. "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." Vergegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 631, 2 USPQ 2d 1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the...claim." Richardson v. Suzuki Motor Co., 868 F.2d 1226, 1236, 9 USPQ 2d 1913, 1920 (Fed. Cir. 1989).

England discloses loading and identifying a digital rights management operating system (DRMOS). Upon power up, a boot loader loads a boot block for a particular operating system. Code in the boot block then loads various drivers and other software components necessary for the OS to function on the computer (England, col. 11, lines 38-45). Once all components are loaded, the OS assumes its identity. A one-way hashing

[16] See Specification page 18, lines 11-19; Figure 2 (reference 270)

function provided by the CPU is used to create a cryptographic digest of all the loaded components. The digest becomes the identity for the OS (England, col. 12, lines 53-58). The DRMOS must provide a secure storage space to protect content permanently stored on the computer by securely storing private keys or session keys for use with encrypted content (England, col. 16, lines 50-55).

England does not disclose, either expressly or inherently, (1) a digest memory to store an isolated digest as recited in claims 1, 21, 41, and 61, (2) a device to attest the isolated execution mode and prove validity of a program loaded into the isolated memory area as recited in claims 1, 21, 41, and 61, (3) a secure environment for an isolated execution mode as recited in claims 1, 21, 41, and 61, (4) a processor operating in one of a normal execution mode and the isolated execution mode as recited in claims 1, 21, 41, and 61, (5) the isolated digest includes at least a digest of one of a processor nub loader, a processor nub, an operating system nub, and a supervisory module loaded in an isolated execution space as recited in claims 2, 22, 42, and 62, (6) an interface to map the device to an address space of a chipset in the secure environment as recited in claims 3, 23, 43, and 63, and (5) a communication storage to exchange security information with the processor in the isolated execution mode as recited in claims 3, 23, 43, and 63.

The Examiner states that the isolated execution mode is interpreted as a mode in which other applications or other unauthorized areas of memory cannot access (Final Office Action, page 3). The Examiner further states that the function of preventing access to a memory while a certain application is running can be interpreted as isolated execution mode because access is prohibited while the trusted application is running in the DRMOS (Final Office Action, page 3). Applicants respectfully disagree for the following reasons.

Claims should be interpreted consistently with the specification, which provides content for the proper construction of the claims because it explains the nature of the patentee's invention. See Renishaw P.L.C. v. Marposs Societa Per Azioni, 158 F.3d 1243 (Fed. Cir. 1998). During patent examination, the pending claims must be "given the broadest reasonable interpretation consistent with the specification". See MPEP 2111. Here, the isolated memory area and the isolated execution mode should be interpreted according to the specification, and not by an arbitrary interpretation. The isolated execution mode is characterized by a number aspects that are not disclosed in England. These aspects include, among other things, division of rings into normal execution ring and

isolated execution ring, OS nub, processor nub, processor nub loader, isolated memory area, isolated execution unit, etc.

England merely discloses creating identities for different versions of a digital right management operating system (DRMOS) (England, col. 11, lines 18-20). The totality of the boot block and the loaded components make up the identity of the operating system (England, col. 11, lines 44-46). England does not disclose an isolated memory area. England merely discloses checks the signature of a component before loading it (England, col. 11, lines 53-54). There is no distinction between an isolated memory area and a normal memory area.

In addition, England merely discloses using a one-way hashing function provided by the CPU to create a cryptographic digest of all the loaded components and use it as the identity of the operating system (England, col. 12, lines 54-58). This is not the same as the digest memory that stores the digest values of the loaded processor nub, the operating system nub, and other supervisory modules loaded into the isolated execution space (See, for example, Specification, page 13, lines 21-24).

Furthermore, England merely discloses a CPU running in a normal mode, not in one of a normal execution mode and an isolated execution mode. When the computer is turned on, the CPU executes a boot loader to load a boot block for a particular operating system (England, col. 11, lines 38-42). In contrast, the isolated execution mode provides a secure environment to the platform. The security features are provided by a number of operations. The isolated execution mode is initialized using a privilege instruction and a processor nub loader (See, for example, Specification, page 7, lines 9-11). The isolated execution mode is supported by an isolated execution circuit including configuration for isolated execution, definition of an isolated area, definition (e.g., decoding and execution) of isolated instructions, etc. (See, for example, Specification, page 10, lines 7-13).

First, as argued above, claims have to be interpreted according to specification. Claim terms are presumed to have the ordinary and customary meanings attributed to them by those of ordinary skill in the art. Sunrace Roots Enter. Co. v. SRAM Corp., 336 F.3d 1298, 1302, 67 USPQ2d 1438, 1441 (Fed. Cir.2003). The ordinary and customary meaning of a term may be evidenced by a variety of sources, Brookhill-Wilk 1, LLC v. Intuitive Surgical, Inc., 334 F.3d 1294, 1298, 67 USPQ2d 1132, 1136 (Fed. Cir. 2003), including: the claims themselves; dictionaries and treatises, Tex. Digital Sys., Inc. v.

Telegenix, Inc., 308 F.3d 1193, 1202, 64 USPQ2d 1812, 1818 (Fed. Cir. 2002); and the written description, the drawings, and the prosecution history, DeMarini Sports, Inc. v. Worth, Inc., 239 F.3d 1314, 1324, 57 USPQ2d 1889, 1894 (Fed. Cir. 2001). Here, the term "isolated execution mode" refers to a mode where the execution is isolated. The term "isolated" defined by the Riverside Webster's II, New College Dictionary, published by Houghton Mifflin Company, in 1995, as: (1) set apart from a group or whole, and (2) placed in quarantine. Therefore, "isolated execution mode" is a mode in which the execution is set apart from the normal execution. This ordinary meaning does not simply involve preventing access to a memory.

Second, an applicant is entitled to be his or her own lexicographer and may rebut the presumption that claim terms are to be given their ordinary and customary meaning by clearly setting forth a definition of the term that is different from its ordinary and customary meaning(s). In re Paulsen, 30 F.3d 1475, 1480, 31 USPQ2d 1671, 1674 (Fed. Cir. 1994). Where an explicit definition is provided by the applicant for a term, that definition will control interpretation of the term as it is used in the claim. Toto Co. v. White Consolidated Industries Inc., 199 F.3d 1295, 1301, 53 USPQ2d 1065, 1069 (Fed. Cir. 1999). Here, applicants defined the term "isolated execution mode" and "normal execution mode" at several places in the specification (See, for example, page 6, lines 6-24, page 10, lines 3-13; page 11, lines 12-22). In particular, an isolated execution mode and a normal execution mode may occupy the same ring in a logical operating architecture. The isolated execution mode is supported by a number of hardware and software elements such as a processor nub, a processor nub loader, OS nub, isolated read/write cycles, etc.

In light of the above, Applicants believe that independent claims 1, 21, 41, and 61, and their respective dependent claims are not anticipated by England.

B. Claims 6-19, 26-39, 46-59, and 66-79 Are Not Obvious over England in view of Ermolovich

In the Final Office Action, the Examiner rejected claims 6-19, 26-39, 46-59, and 66-79 under 35 U.S.C. §103(a) as being unpatentable over England in view of U.S. Patent No. 4,319,323 issued to Ermolovich ("Ermolovich"). Applicants respectfully traverse the rejection and contend that the Examiner has not met the burden of establishing a prima facie case of obviousness.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. *MPEP §2143, p. 2100-129 (8th Ed., rev. 2, May 2004).* Applicants respectfully contend that there is no suggestion or motivation to combine their teachings, and thus no *prima facie* case of obviousness has been established.

England discloses loading and identifying a digital rights management operating system as discussed above.

Ermolovich discloses a communications device for data processing system. A device status is built and inserted into a packet as a status longword before inserting a command packet into a termination queue (Ermolovich, col. 85, lines 37-41). The device status contains the status of a communication device after the communication device processes a command packet (Ermolovich, col. 13, lines 37-43). A command interpreter transfers contents of a command field to a command register in an external device (Ermolovich, col. 12, lines 2-6). The communication device may directly write to or read from buffers in the data block and command block (Ermolovich, col. 7, lines 54-58).

England and Ermolovich, taken alone or in any combination, do not disclose, suggest, or render obvious (1) a communication storage to exchange security information with the processor in the isolated execution mode, (2) a status register to store device status of the device, (3) a command register to store a device command for a command interface set; and (4) an input/output block (IOB) to store input and output data corresponding to the command.

There is no motivation to combine England and Ermolovich because neither of them addresses the problem of isolated execution. There is no teaching or suggestion that a digest memory, a device to attest isolated execution mode, and a processor having normal and isolated execution modes is present. England, read as a whole, does not suggest the desirability of attesting an isolated execution mode, or proving validity of a program, or a configuration storage in a communication storage corresponding to an address space for an isolated execution mode. England does not disclose or suggest an isolated execution mode as discussed above. Ermolovich merely discloses status word in a command packet for a

communication device, not a configuration storage for an isolated execution mode. Ermolovich merely discloses a state to initiate a data transfer. In this state, a command interpreter is enabled to transfer the contents of the command field to a command register in the external device (Ermolovich, col. 12, lines 2-6). As noted above, the command register here is used only for communication devices and data transfers, not to allow the attestation key memory device to exchange security information with at least one processor. The Examiner further states that Ermolovich discloses an input/output block to store input and output data and cites column 71, lines 40-64 (Final Office Action, page 6). However, the cited paragraph merely discloses a data block and command block which contain buffers to/from which the communication device directly writes/reads (Ermolovich, col. 71, lines 54-59). This is not the same as input and output data corresponding to the command used in exchanging security information and corresponding to an address space of a chipset in a secure environment.

The Examiner failed to establish a prima facie case of obviousness and failed to show there is teaching, suggestion or motivation to combine the references. "When determining the patentability of a claimed invention which combined two known elements, 'the question is whether there is something in the prior art as a whole suggest the desirability, and thus the obviousness, of making the combination.'" In re Beattie, Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co., 730 F.2d 1452, 1462, 221 USPQ (BNA) 481, 488 (Fed. Cir. 1984). "To support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or implicitly suggest the claimed invention or the Examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the references." Ex parte Clapp, 227 USPQ 972, 973. (Bd.Pat.App.&Inter. 1985).

In the present invention, the cited references do not expressly or implicitly suggest (1) a communication storage to exchange security information with the processor in the isolated execution mode, (2) a status register to store device status of the device, (3) a command register to store a device command for a command interface set; and (4) an input/output block (IOB) to store input and output data corresponding to the command. In addition, the Examiner failed to present a convincing line of reasoning as to why a

combination of England and Ermolovich is an obvious application of attestation using an isolated digest and an isolated execution mode.

VIII. CONCLUSION

Applicant respectfully requests that the Board enter a decision overturning the Examiner's rejection of all pending claims, and holding that the claims are neither anticipated nor rendered obvious by the prior art.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: November 25, 2005

THINH V. NGUYEN
Reg. No. 42,034

12400 Wilshire Blvd., 7th Floor
Los Angeles, CA 90025-1026
(714) 557-3800

IX. CLAIMS APPENDIX

The claims of the present application which are involved in this appeal are as follows:

1. (previously presented) An apparatus comprising:

a digest memory to store an isolated digest in a secure environment for an isolated execution mode, the secure environment being associated with an isolated memory area accessible by at least one processor, the at least one processor operating in one of a normal execution mode and the isolated execution mode; and

an attestation key memory (AKM) device coupled to the digest memory to attest the isolated execution mode and prove validity of a program loaded into the isolated memory area using the isolated digest.

2. (previously presented) The apparatus of claim 1 wherein the isolated digest includes at least a digest of one of a processor nub loader, a processor nub, an operating system nub, and a supervisory module loaded in an isolated execution space.

3. (previously presented) The apparatus of claim 2 further comprising:

an interface to map the device to an address space of a chipset in the secure environment; and

a communication storage corresponding to the address space to allow the AKM device to exchange security information with the at least one processor, the security information including at least one of a static public key and a static key certificate.

4. (original) The apparatus of claim 3 wherein the device accesses a chipset storage via the address space.

5. (original) The apparatus of claim 4 wherein the communication storage comprises:

a configuration storage to store device configuration information.

6. (original) The apparatus of claim 5 wherein the communication storage further comprises:

a status register to store device status of the device;

a command register to store a device command for a command interface set; and

an input/output block (IOB) to store input and output data corresponding to the command.

7. (original) The apparatus of claim 6 wherein the configuration storage comprises:

a public key storage to store the static public key;

a key certificate storage to store the static key certificate; and

an interface set storage to store an interface set identifier, the interface set identifier identifying a command interface set supported by the device.

8. (original) The apparatus of claim 7 wherein the configuration storage further comprises:

a manufacturer identifier storage to store a manufacturer identifier; and

a revision storage to store a revision identifier.

9. (original) The apparatus of claim 7 wherein the command interface set is an initialization set, the initialization set supporting a reset command and a connect command.

10. (original) The apparatus of claim 7 wherein the command interface set is an attestation set, the attestation set performing at least one of a public key enumeration, a key certificate enumeration, and a signing operation.

11. (original) The apparatus of claim 10 wherein the status register comprises:

a connection field to provide a connection status to indicate that the device is responsive to the connect command; and

an estimate field to provide an estimate of processing time for an operation specified in the command.

12. (original) The apparatus of claim 11 wherein the status register further comprises:

a self-test field to indicate status of a self test in response to the reset command.

13. (original) The apparatus of claim 10 wherein the public key enumeration enumerates an additional public key other than the static public key.

14. (original) The apparatus of claim 10 wherein the key certificate enumeration enumerates an additional key certificate other than the static key certificate.

15. (original) The apparatus of claim 10 wherein the sign operation generates a signature to attest validity of the secure environment using a private key provided by the chipset.

16. (original) The apparatus of claim 15 wherein the signature corresponds to signing a chipset parameter.

17. (previously presented) The apparatus of claim 16 wherein the chipset parameter is one of a processor nub loader hash, a chipset hash log, a software hash, and a nonce.

18. (previously presented) The apparatus of claim 17 wherein the processor nub loader hash and the chipset hash log are stored in the chipset storage.

19. (previously presented) The apparatus of claim 18 wherein the software hash and the nonce are provided by a processor nub.

20. (original) The apparatus of claim 3 wherein the device accesses a remote server via the address space.

21. (previously presented) A method comprising:

storing an isolated digest in a digest memory in a secure environment for an isolated execution mode, the secure environment being associated with an isolated memory area accessible by at least one processor, the at least one processor operating in one of a normal execution mode and the isolated execution mode; and

attesting the isolated execution mode and proving validity of a program loaded into the isolated memory area using an attestation key memory (AKM) device and the isolated digest.

22. (previously presented) The method of claim 21 wherein the isolated digest includes at least a digest of one of a processor nub loader, a processor nub, an operating system nub, and a supervisory module loaded in an isolated execution space.

23. (previously presented) The method of claim 22 further comprising:

mapping the AKM device to an address space of a chipset in the same environment; and

exchanging security information between the AKM device and the at least one processor via a communication storage corresponding to the address space, the security information including at least one of a static public key and a static key certificate.

24. (original) The method of claim 23 wherein the device accesses a chipset storage via the address space.

25. (original) The method of claim 24 wherein exchanging comprises:

storing device configuration information in a configuration storage.

26. (original) The method of claim 25 wherein exchanging further comprises:

storing device status of the device in a status register;

performing a device command corresponding to a command interface set to a command register; and

storing input and output data corresponding to the command in an input/output block (IOB).

27. (original) The method of claim 26 wherein storing in the configuration storage comprises:

storing the static public key in a public key storage;

storing the static key certificate in a key certificate storage; and

storing an interface set identifier in an interface set storage, the interface set identifier identifying a command interface set supported by the device.

28. (original) The method of claim 27 wherein storing in the configuration storage further comprises:

storing a manufacturer identifier in a manufacturer identifier storage; and

storing a revision identifier in a revision storage.

29. (original) The method of claim 27 wherein performing the device command comprises performing a reset command and a connect command corresponding to an initialization set.

30. (original) The method of claim 27 wherein performing the device command comprises performing at least one of a public key enumeration, a key certificate enumeration, and a signing operation, the public key enumeration, the key certificate enumeration, and the signing operation corresponding to an attestation set.

31. (original) The method of claim 30 wherein storing the device status comprises:

providing a connection status to indicate that the device is responsive to the connect command; and

providing an estimate of processing time for an operation specified in the command.

32. (original) The method of claim 31 wherein storing the device status further comprises:

indicating status of a self test in response to the reset command.

33. (original) The method of claim 30 wherein performing the public key enumeration comprises enumerating an additional public key other than the static public key.

34. (original) The method of claim 30 wherein performing the key certificate enumeration comprises enumerating an additional key certificate other than the static key certificate.

35. (original) The method of claim 30 wherein performing the sign operation comprises generating a signature to attest validity of the secure environment using a private key provided by the chipset.

36. (original) The method of claim 35 wherein the signature corresponds to signing a chipset parameter.

37. (previously presented) The method of claim 36 wherein the chipset parameter is one of a processor nub loader hash, a chipset hash log, a software hash, and a nonce.

38. (previously presented) The method of claim 37 wherein the processor nub loader hash and the chipset hash log are stored in the chipset storage.

39. (previously presented) The method of claim 38 wherein the software hash and the nonce are provided by a processor nub.

40. (original) The method of claim 23 wherein the device accesses a remote server via the address space.

41. (previously presented) A computer program product comprising:

a machine readable medium having program code embedded therein, the computer program product comprising:

computer readable program code to store an isolated digest in a digest memory in a secure environment for an isolated execution mode, the secure environment being

associated with an isolated memory area accessible by at least one processor, the at least one processor operating in one of a normal execution mode and the isolated execution mode; and

computer readable program code to attest the isolated execution mode and proving validity of a program loaded into the isolated memory area using an attestation key memory (AKM) device and the isolated digest.

42. (previously presented) The computer program product of claim 41 wherein the isolated digest includes at least a digest of one of a processor nub loader, a processor nub, an operating system nub, and a supervisory module loaded in an isolated execution space.

43. (previously presented) The computer program product of claim 42 wherein the computer program product further comprising:

computer readable program code to map the AKM device to an address space of a chipset; and

computer readable program code to exchange security information between the AKM device and the at least one processor via a communication storage corresponding to the address space, the security information including at least one of a static public key and a static key certificate.

44. (original) The computer program product of claim 43 wherein the device accesses a chipset storage via the address space.

45. (previously presented) The computer program product of claim 44 wherein the computer readable program code to exchange comprises:

computer readable program code to store device configuration information in a configuration storage.

46. (previously presented) The computer program product of claim 45 wherein the computer readable program code to exchange further comprises:

computer readable program code to store device status of the device in a status register;

computer readable program code to perform a device command corresponding to a command interface set to a command register; and

computer readable program code to store input and output data corresponding to the command in an input/output block (IOB).

47. (previously presented) The computer program product of claim 46 wherein the computer readable program code to store in the configuration storage comprises:

computer readable program code to store the static public key in a public key storage;

computer readable program code to store the static key certificate in a key certificate storage; and

computer readable program code to store an interface set identifier in an interface set storage, the interface set identifier identifying a command interface set supported by the device.

48. (previously presented) The computer program product of claim 47 wherein the computer readable program code to store in the configuration storage further comprises:

computer readable program code to store a manufacturer identifier in a manufacturer identifier storage; and

computer readable program code to store a revision identifier in a revision storage.

49. (previously presented) The computer program product of claim 47 wherein the computer readable program code to perform the device command comprises computer readable program code to perform a reset command and a connect command corresponding to an initialization set.

50. (previously presented) The computer program product of claim 47 wherein the computer readable program code for to perform the device command comprises computer readable program code to perform at least one of a public key enumeration, a key certificate enumeration, and a signing operation, the public key enumeration, the key certificate enumeration, and the signing operation corresponding to an attestation set.

51. (previously presented) The computer program product of claim 50 wherein the computer readable program code to store the device status comprises:

computer readable program code to provide a connection status to indicate that the device is responsive to the connect command; and

computer readable program code to provide an estimate of processing time for an operation specified in the command.

52. (previously presented) The computer program product of claim 51 wherein the computer readable program code to store the device status further comprises:

computer readable program code to indicate status of a self test in response to the reset command.

53. (previously presented) The computer program product of claim 50 wherein the computer readable program code to perform the public key enumeration comprises computer readable program code to enumerate an additional public key other than the static public key.

54. (previously presented) The computer program product of claim 50 wherein the computer readable program code to perform the key certificate enumeration comprises computer readable program code to enumerate an additional key certificate other than the static key certificate.

55. (previously presented) The computer program product of claim 50 wherein the computer readable program code to perform the sign operation comprises computer readable program code to generate a signature to attest validity of the secure environment using a private key provided by the chipset.

56. (original) The computer program product of claim 55 wherein the signature corresponds to signing a chipset parameter.

57. (previously presented) The computer program product of claim 56 wherein the chipset parameter is one of a processor nub loader hash, a chipset hash log, a software hash, and a nonce.

58. (previously presented) The computer program product of claim 57 wherein the processor nub loader hash and the chipset hash log are stored in the chipset storage.

59. (previously presented) The computer program product of claim 58 wherein the software hash and the nonce are provided by a processor nub.

60. (original) The computer program product of claim 43 wherein the device accesses a remote server via the address space.

61. (previously presented) A system comprising:

an attestation key memory (AKM) device;

at least one processor operating in a secure environment, the at least one processor having one of a normal execution mode and an isolated execution mode;

a memory coupled to the at least one processor, the memory having an isolated memory area accessible to the at least one processor in the isolated execution mode; and

a chipset coupled to the at least one processor and the memory, the chipset having a circuit, the circuit comprising:

a digest memory to store an isolated digest used with the device to attest the isolated execution mode and prove validity of a program loaded into the isolated memory area.

62. (previously presented) The system of claim 61 wherein the isolated digest includes at least a digest of one of a processor nub loader, a processor nub, an operating system nub, and a supervisory module loaded in an isolated execution space.

63. (previously presented) The system of claim 62 wherein the circuit further comprises:

an interface to map the device to an address space of the chipset; and

a communication storage corresponding to the address space to allow the AKM device to exchange security information with the at least one processor, the security information including at least one of a static public key and a static key certificate.

64. (original) The system of claim 63 wherein the device accesses a chipset storage via the address space.

65. (original) The system of claim 64 wherein the communication storage comprises:

a configuration storage to store device configuration information.

66. (original) The system of claim 65 wherein the communication storage further comprises:

a status register to store device status of the device;

a command register to store a device command for a command interface set; and

an input/output block (IOB) to store input and output data corresponding to the command.

67. (original) The system of claim 66 wherein the configuration storage comprises:

a public key storage to store the static public key;

a key certificate storage to store the static key certificate; and

an interface set storage to store an interface set identifier, the interface set identifier identifying a command interface set supported by the device.

68. (original) The system of claim 67 wherein the configuration storage further comprises:

a manufacturer identifier storage to store a manufacturer identifier; and

a revision storage to store a revision identifier.

69. (original) The system of claim 67 wherein the command interface set is an initialization set, the initialization set supporting a reset command and a connect command.

70. (original) The system of claim 67 wherein the command interface set is an attestation set, the attestation set performing at least one of a public key enumeration, a key certificate enumeration, and a signing operation.

71. (original) The system of claim 70 wherein the status register comprises:

a connection field to provide a connection status to indicate that the device is responsive to the connect command; and

an estimate field to provide an estimate of processing time for an operation specified in the command.

72. (original) The system of claim 71 wherein the status register further comprises:

a self-test field to indicate status of a self test in response to the reset command.

73. (original) The system of claim 70 wherein the public key enumeration enumerates an additional public key other than the static public key.

74. (original) The system of claim 70 wherein the key certificate enumeration enumerates an additional key certificate other than the static key certificate.

75. (original) The system of claim 70 wherein the sign operation generates a signature to attest validity of the secure environment using a private key provided by the chipset.

76. The system of claim 75 wherein the signature corresponds to signing a chipset parameter.

77. (previously presented) The system of claim 76 wherein the chipset parameter is one of a processor nub loader hash, a chipset hash log, a software hash, and a nonce.

78. (previously presented) The system of claim 77 wherein the processor nub loader hash and the chipset hash log are stored in the chipset storage.

79. (previously presented) The system of claim 78 wherein the software hash and the nonce are provided by a processor nub.

80. (original) The system of claim 63 wherein the device accesses a remote server via the address space.

X. EVIDENCE APPENDIX

None

XI. RELATED PROCEEDINGS APPENDIX

None